พุฒิพงศ์ นพคุณ

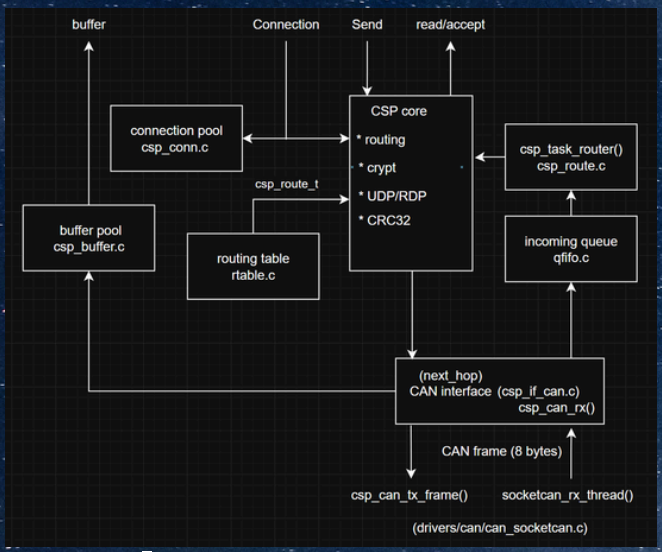
CUBESAT SPACE PROTOCOL

Cubesat Space Protocol คืออะไร?

Cubesat Space Protocol (CSP) เป็น protocol stack ขนาดเล็กที่เขียนด้วยภาษา C โดย CSP ออกแบบมาเพื่อการสื่อสารระหว่าง embedded system ในเครือข่ายขนาดเล็ก เช่น ดาวเทียม Cubesat, microsatellite แนวคิดคล้าย TCP/IP model แต่ตัดส่วนที่หนักและซับซ้อนออก เรียกได้ว่าเป็น **"ตัวกลางที่ให้ subsystem ต่าง ๆ คุยภาษาเดียวกัน"**
โดย CSP รองรับ routing, transport protocol, socket API, packet management, multi-node communication, multiple interfaces (CAN, UART, I2C, Ethernet ฯลฯ) ทั้งหมดนี้ใช้ RAM และ CPU ต่ำมาก เหมาะกับ Microcontroller ขนาดเล็ก

การทำงานภายใน CSP เป็นอย่างไร?

Data Flow (ตัวอย่างจาก CAN interface)



การทำงานของ data flow
- Application สร้าง connection
- ขอ packet buffer
- ใส่ payload
- เรียก csp_send()
- Router หาเส้นทาง
- Interface ส่งผ่าน CAN/UART/I2C

CSP Header 1.x

Bit offset	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	Priority				Source				Destination				Destination Port						Source Port						Reserved				HMAC	XTEA	RDP	CRC
32	Data (0 – 65,535 bytes)																															

ความหมายของแต่ละฟิลด์
CSP identifier:
1. Priority — 2 bits
   - ใช้กำหนดลำดับความสำคัญในคิวส่ง
   - Router จะส่ง packet ที่มี priority สูงกว่าก่อนเสมอ
2. Source Address — 5 bits
   - หมายเลข node ต้นทางใน CSP network
3. Destination Address— 5 bits
   - หมายเลข node ปลายทาง
4. Destination Port — 6 bits
   - คล้าย TCP/UDP port สำหรับเรียกใช้ service
5. Source Port — 6 bits
   - Port ของผู้ส่ง ใช้สำหรับ reply กลับ

flags:
1. Reversed — 4 bits
   - 3 bit ยังไม่ได้มีการใช้งาน
   - 1 bit สำหรับการแตกข้อมูล (fragmentation)
2. HMAC verification
3. XTEA encryption
4. RDP ใช้ Reliable Datagram Protocol
5. CRC32 checksum

Application

ตัวอย่างการใช้งานจริง
OBC = node 1, EPS = node 2
OBC ต้องการส่ง "จ่ายไฟ 12 V" ไปหา EPS
Step 1: Application Layer
ข้อมูลจะเป็น: [Payload]
Step 2: CSP Header
CSP จะเพิ่ม header เข้าไปกลายเป็น
[CSP Header][Payload]
กลายเป็น packet พร้อมวิ่งใน network แล้ว
Step 3: Router
CSP Router จะคิดว่าจะส่ง packet นี้ออก interface ไหน?
เช่น: Node 2 อยู่บน CAN1 Router จึงส่ง packet ไป interface
Step 4: CAN Interface
แบ่ง packet เป็น CAN frame เพราะ CAN ส่งได้แค่ 8 bytes
CSP จะทำการแบ่ง Packet -> Frame 1, Frame 2, Frame 3
Step 5: CAN Driver
ตอนนี้ CAN interface จะเรียก CAN Driver
แล้ว hardware จะส่งออก bus จริง
Step 6: CAN Bus
ตอนนี้ข้อมูลวิ่งบนสาย CAN จริง
MCU1 -----> MCU2
ทุก node บน bus จะเห็น frame นี้
แต่มีเฉพาะ node 2 ที่ระบุตรงกันเท่านั้นที่จะรับ
ส่วนฝั่งรับก็จะทำคล้าย ๆ กันย้อนจาก step 6 ขึ้นไป step 1
(step 4 จะรวม Frame เป็น Packet)